RTC overview

Contents

Stable: 05.11.2021 - 08:57 / Revision: 05.11.2021 - 08:55

Stable: 08.11.2021 - 08.01 | Revision: 08.11.2021 - 08.08

A quality version of this page, approved on *14 September 2021*, was based off this revision.

This article gives information about the Linux® RTC framework. The RTC framework is involved in precise time countdown.

Contents

1 Framework purpose

The RTC keeps the system time up to date and can wake up the system from the standby power mode at a programmed time.

A general presentation of the RTC framework is available in the Linux RTC documentation [1].

2 System overview

RTC framework overview

2.1 Component description

- **RTC (hardware)**
  RTC dedicated hardware block in STM32MPU product.
- **rtc-stm32**
  RTC ST driver.
- **rtc**
  Linux RTC framework, it provides API to RTC driver and interfaces to user.
- **Sysfs interface**
  Sysfs interface is accessible via /sys/class/rtc/rtcX/.
- **Procfs interface**
  Procfs interface is accessible via /proc/driver/rtc.
- **Char device interface**
  Device interface is accessible via /dev/rtcX.
- **User application**
  The user application can be an user built application or an community application like hwclock.

2.2 API description

API is described in Linux RTC documentation [1].

3 Configuration

3.1 Kernel Configuration

Activate **rtc-stm32** driver in kernel configuration using the Linux Menuconfig tool: Menuconfig or how to configure kernel

```
Device drivers  --->
    -*- Real Time Clock  --->
        <*> STM32 RTC
```

3.2 Device tree configuration

Please refer to the RTC device tree configuration.

4 How to use the framework

Please refer to the How to use the RTC.

5 How to trace and debug the framework

5.1 How to trace

Dynamic debug traces can be added using the following commands:

```
echo -n 'file rtc-stm32.c +p'>/sys/kernel/debug/dynamic_debug/control
```

6 Source code location

- rtc-stm32: drivers/rtc/rtc-stm32.c
- api: include/linux/rtc.h
- framework:
  - drivers/rtc/class.c
  - drivers/rtc/dev.c
  - drivers/rtc/interface.c
  - drivers/rtc/lib.c
  - drivers/rtc/proc.c
  - drivers/rtc/sysfs.c
- device-tree bindings constants: include/dt-bindings/rtc/rtc-stm32.h

7 References

- 1.01.1 Linux RTC documentation

Stable: 09.09.2021 - 14:00 / Revision: 07.09.2021 - 14:49

This article gives information about the Linux® RTC framework. The RTC framework is involved in precise time countdown.

Contents

1 Framework purpose

The RTC keeps the system time up to date and can wake up the system from the standby power mode at a programmed time.

A general presentation of the RTC framework is available in the Linux RTC documentation [1].

2 System overview

RTC framework overview

2.1 Component description

- **RTC (hardware)**
  RTC dedicated hardware block in STM32MPU product.
- **rtc-stm32**
  RTC ST driver.
- **rtc**
  Linux RTC framework, it provides API to RTC driver and interfaces to user.
- **Sysfs interface**
  Sysfs interface is accessible via /sys/class/rtc/rtcX/.
- **Procfs interface**
  Procfs interface is accessible via /proc/driver/rtc.
- **Char device interface**
  Device interface is accessible via /dev/rtcX.
- **User application**
  The user application can be an user built application or an community application like hwclock.

2.2 API description

API is described in Linux RTC documentation [1].

3 Configuration

3.1 Kernel Configuration

Activate **rtc-stm32** driver in kernel configuration using the Linux Menuconfig tool: Menuconfig or how to configure kernel

```
Device drivers  --->
    -*- Real Time Clock  --->
        <*> STM32 RTC
```

3.2 Device tree configuration

Please refer to the RTC device tree configuration.

4 How to use the framework

Please refer to the How to use the RTC.

5 How to trace and debug the framework

5.1 How to trace

Dynamic debug traces can be added using the following commands:

```
echo -n 'file rtc-stm32.c +p'>/sys/kernel/debug/dynamic_debug/control
```

6 Source code location

- rtc-stm32: drivers/rtc/rtc-stm32.c
- api: include/linux/rtc.h
- framework:
  - drivers/rtc/class.c
  - drivers/rtc/dev.c
  - drivers/rtc/interface.c
  - drivers/rtc/lib.c
  - drivers/rtc/proc.c
  - drivers/rtc/sysfs.c
- device-tree bindings constants: include/dt-bindings/rtc/rtc-stm32.h

7 References

- 1.01.1 Linux RTC documentation

Stable: 31.03.2021 - 08:47 / Revision: 26.03.2021 - 08:44

This article gives information about the Linux® RTC framework. The RTC framework is involved in precise time countdown.

Contents

1 Framework purpose

The RTC keeps the system time up to date and can wake up the system from the standby power mode at a programmed time.

A general presentation of the RTC framework is available in the Linux RTC documentation [1].

2 System overview

RTC framework overview

2.1 Component description

- **RTC (hardware)**
  RTC dedicated hardware block in STM32MPU product.
- **rtc-stm32**
  RTC ST driver.
- **rtc**
  Linux RTC framework, it provides API to RTC driver and interfaces to user.
- **Sysfs interface**
  Sysfs interface is accessible via /sys/class/rtc/rtcX/.
- **Procfs interface**
  Procfs interface is accessible via /proc/driver/rtc.
- **Char device interface**
  Device interface is accessible via /dev/rtcX.
- **User application**
  The user application can be an user built application or an community application like hwclock.

2.2 API description

API is described in Linux RTC documentation [1].

3 Configuration

3.1 Kernel Configuration

Activate **rtc-stm32** driver in kernel configuration using the Linux Menuconfig tool: Menuconfig or how to configure kernel

```
Device drivers  --->
    -*- Real Time Clock  --->
        <*> STM32 RTC
```

3.2 Device tree configuration

Please refer to the RTC device tree configuration.

4 How to use the framework

Please refer to the How to use the RTC.

5 How to trace and debug the framework

5.1 How to trace

Dynamic debug traces can be added using the following commands:

```
echo -n 'file rtc-stm32.c +p'>/sys/kernel/debug/dynamic_debug/control
```

6 Source code location

- rtc-stm32: drivers/rtc/rtc-stm32.c
- api: include/linux/rtc.h
- framework:
  - drivers/rtc/class.c
  - drivers/rtc/dev.c
  - drivers/rtc/interface.c
  - drivers/rtc/lib.c
  - drivers/rtc/proc.c
  - drivers/rtc/sysfs.c
- device-tree bindings constants: include/dt-bindings/rtc/rtc-stm32.h

7 References

- 1.01.1 Linux RTC documentation

Stable: 09.09.2021 - 14:07 / Revision: 09.09.2021 - 14:07

This article gives information about the Linux® RTC framework. The RTC framework is involved in precise time countdown.

Contents

1 Framework purpose

The RTC keeps the system time up to date and can wake up the system from the standby power mode at a programmed time.

A general presentation of the RTC framework is available in the Linux RTC documentation [1].

2 System overview

RTC framework overview

2.1 Component description

- **RTC (hardware)**
  RTC dedicated hardware block in STM32MPU product.
- **rtc-stm32**
  RTC ST driver.
- **rtc**
  Linux RTC framework, it provides API to RTC driver and interfaces to user.
- **Sysfs interface**
  Sysfs interface is accessible via /sys/class/rtc/rtcX/.
- **Procfs interface**
  Procfs interface is accessible via /proc/driver/rtc.
- **Char device interface**
  Device interface is accessible via /dev/rtcX.
- **User application**
  The user application can be an user built application or an community application like hwclock.

2.2 API description

API is described in Linux RTC documentation [1].

3 Configuration

3.1 Kernel Configuration

Activate **rtc-stm32** driver in kernel configuration using the Linux Menuconfig tool: Menuconfig or how to configure kernel

```
Device drivers  --->
    -*- Real Time Clock  --->
        <*> STM32 RTC
```

3.2 Device tree configuration

Please refer to the RTC device tree configuration.

4 How to use the framework

Please refer to the How to use the RTC.

5 How to trace and debug the framework

5.1 How to trace

Dynamic debug traces can be added using the following commands:

```
echo -n 'file rtc-stm32.c +p'>/sys/kernel/debug/dynamic_debug/control
```

6 Source code location

- rtc-stm32: drivers/rtc/rtc-stm32.c
- api: include/linux/rtc.h
- framework:
  - drivers/rtc/class.c
  - drivers/rtc/dev.c
  - drivers/rtc/interface.c
  - drivers/rtc/lib.c
  - drivers/rtc/proc.c
  - drivers/rtc/sysfs.c
- device-tree bindings constants: include/dt-bindings/rtc/rtc-stm32.h

7 References

- [1.01.1] Linux RTC documentation